

Run Time Analysis for Random Local Search on Generalized Majority Functions

Carola Doerr and Martin S. Krejca

Abstract—Run time analysis of evolutionary algorithms recently makes significant progress in linking algorithm performance to *algorithm* parameters. However, settings that study the impact of *problem* parameters are rare. The recently proposed W-model provides a good framework for such analyses, generating pseudo-Boolean optimization problems with *tunable* properties.

We initiate theoretical research of the W-model by studying how one of its properties—*neutrality*—influences the run time of random local search. Neutrality creates plateaus in the search space by first performing a majority vote for subsets of the solution candidate and then evaluating the smaller-dimensional string via a low-level fitness function.

We prove upper bounds for the expected run time of random local search on this MAJORITY problem for its entire parameter spectrum. To this end, we provide a theorem, applicable to many optimization algorithms, that links the run time of MAJORITY with its symmetric version HASMAJORITY, where a sufficient majority is needed to optimize the subset. We also introduce a generalized version of classic drift theorems as well as a generalized version of Wald’s equation, both of which we believe to be of independent interest.

Index Terms—evolutionary computation; run time analysis; plateau; neutrality; majority

I. INTRODUCTION

Randomized search heuristics, such as evolutionary algorithms (EAs), have been applied with great success to real-world optimization problems for which the user is faced with limited information about the problem or limited resources to solve it. This is an impressive feat, since such problems are typically hard due to a combination of different challenging features. In order to better understand the reasons behind why EAs perform so well in such settings, theoretical results analyze benchmark functions that encapsulate specific features assumed to be also present in real-world problems. At present, these functions are typically unrelated to each other, and it is difficult to derive how they interact with one another. This makes it hard to translate such theoretical results to more realistic problems. An alternative approach for deriving theoretical results is to consider a class of different functions that can be combined in well-defined ways, allowing to provide guarantees for more complex problems. The recently introduced W-model [1] provides such function classes. Given a pseudo-Boolean function f , the W-model proposes a sequence of four

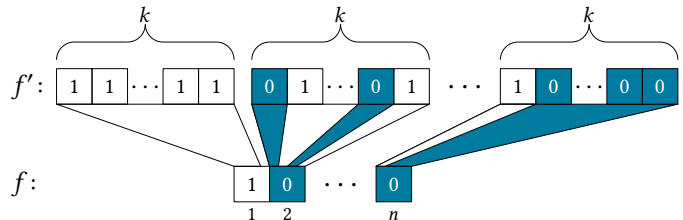


Fig. 1: Given a pseudo-Boolean function f defined over bit strings of length $n \in \mathbb{N}_{>0}$ and a parameter $k \in \mathbb{N}_{>0}$, *neutrality* creates a function f' defined over bit strings of length nk , consisting of n blocks of length k each. When evaluating f' , first, for each of the k blocks, the majority of its bits is determined. Then, f is evaluated on the resulting bit string of length n of the majority outcomes.

independent steps that each change a different feature of f . The degree by which each feature is changed is regulated by parameters specific to each step. Thus, overall, the W-model allows to create a diverse set of functions, all based on f , that showcases various features in different intensities.

In this article, we initiate the run time analysis of the W-model. To this end, we focus on one of its features called *neutrality*. Given a function f defined over bit strings of length n as well as an integer parameter k , neutrality substitutes each bit of a length- n bit string by k bits. When evaluating a new bit string of length nk , each of the subsequent blocks of k bits is reduced to the majority among its bit values, and the resulting bit string of length n is evaluated by f (see Figure 1). Overall, neutrality introduces a two-stage process into the optimization of f . On a high level, f is optimized. On a low level, the correct majority value of each block is determined. Assuming that the different blocks are independent, analyzing the run times of the two stages separately and then combining these results leads to a run time result for f .

Doerr et al. [2] analyzed the high-level approach of the aforementioned two-stage process diligently for the class of *separable* functions, where a function is separable if its value is the sum of the values of n independent sub-functions. The authors show that the run time of random local search (RLS) and of the $(1 + 1)$ EA is, in expectation, bounded from above by the slowest expected run time of a sub-function times $\ln(n)$. The logarithmic overhead is a result of the classical coupon collector problem that accounts for optimizing *all* n sub-functions correctly.

The second stage of the two-stage optimization process sketched above concerns analyzing the run time of a majority

C. Doerr is with Sorbonne Université, CNRS, LIP6, Paris, France
Martin S. Krejca is with LIX, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, Palaiseau, France.

This work was financially supported by the Paris Île-de-France Region via the DIM RFSI AlgoSelect project and via the European Union’s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 945298-ParisRegionFP.

function. Since the value returned by such a function only changes if the majority of the bits in its input changes, this leads to analyzing the behavior of an algorithm in a landscape of equal fitness, called a *plateau*. Changing the number of bits to the optimal majority requires an EA to perform a random walk on the plateau until finding the optimal majority, at which point in time the block is optimized. We refer to this search behavior as *crossing* the plateau.

In order to better understand the impact of the plateau size on the optimization process, we extend the notion of majority for the W-model. To this end, we introduce the function HASMAJORITY_r , defined over bit strings of length $n \in \mathbb{N}^+$ (where n is even¹), having a parameter $r \in [0, n/2] \cap \mathbb{N}$ how strong the majority has to be in order to be counted. HASMAJORITY_r takes the two values 0 and 1. A bit string has a HASMAJORITY_r -value of 1 if its number of 0s or its number of 1s is at least $n/2+r$, and it has a value of 0 otherwise. We also introduce the asymmetric version MAJORITY_r , where a bit string only has a function value of 1 if it has at least $n/2+r$ 1s (and it is 0 otherwise).

a) *Our results:* We analyze RLS, which iteratively constructs new solutions by flipping exactly one bit, chosen uniformly at random, in its currently best solution. We bound the expected run time of RLS, that is, the number of iterations until an optimal solution is found for the first time, on HASMAJORITY and MAJORITY from above for the entire range of r . Using the results by Doerr et al. [2], this implies run time bounds for ONEMAX with added neutrality (Corollary 13). We first analyze the symmetric case of HASMAJORITY (similar to the result by Bian et al. [3]) and show with Theorem 6 the complex dependency of the run time of RLS on the parameter r , resulting in multiple different regimes (Figure 2). For values of r constant in n , the run time is constant, and it grows quadratically in r until $r = O(\sqrt{n})$. For $r = \omega(\sqrt{n}) \cap o(n)$, the expected run time is mainly given by an exponential in r^2/n . For the remaining regime of r , the expected run time is at least exponential. The behavior for MAJORITY is similar, as Theorem 7 shows. The main difference is an overhead of $n \ln(r)$, which is caused by the larger size of the plateau. For the special case $r = n/2 - 1$, the MAJORITY function reduces to the well-known NEEDLE function, for which the run time of RLS, the $(1+1)$ EA (EA), and a number of generalizations are very well understood [4], [5].

Our result for MAJORITY is based on a restart argument applied to HASMAJORITY . Since such restart arguments are not specific to the context of EAs or even optimization, we analyze the situation in a general setting (Theorem 8). Our result applies to any assortment of interleaved stopping times. We prove Theorem 8 via a generalized version of Wald's equation (Theorem 5), which we could not find in this detail and generality in the literature. We prove the generalized version of Wald's equation via a new general basic drift theorem (Theorem 4), which relates the expected progress of a random process more explicitly to the values at the start and at the end of the process than traditional drift theorems. As all of these

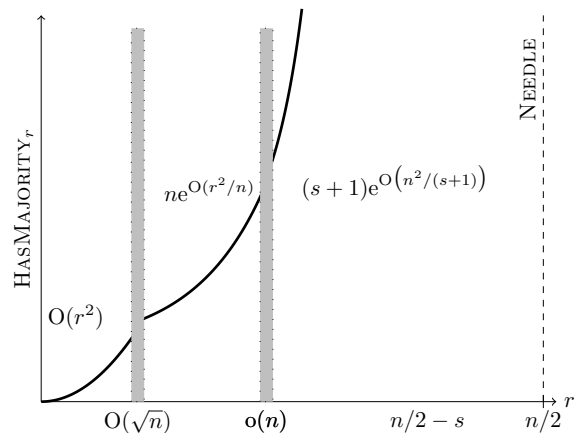


Fig. 2: The dependency of the expected run time of RLS on HASMAJORITY_r with respect to the majority parameter r as proven in Theorem 6. The gray boxes denote regions in which the asymptotics switch (sharply) from $O(\sqrt{n})$ to $\omega(\sqrt{n})$ and from $o(n)$ to $\Theta(n)$. The value $s \in \mathbb{N}$ is any value such that there is a $c \in \Theta(1)$ with $c < 1/2$ such that $s \leq cn$. The dashed line at $n/2$ denotes the run time on NEEDLE .

theorems relate to scenarios commonly found in the analysis of EAs, we are confident that they are of independent interest.

With Corollary 9, we explicitly connect the expected run times of HASMAJORITY and MAJORITY for a large class of EAs, providing a general framework for our setting. This result comes with multiple conditions, due to its generality. However, we also provide additional statements (Lemmas 10 and 11) that show that many EAs satisfy the conditions of Corollary 9. In order to apply Corollary 9 to any EA we cover with this framework, it then remains to provide bounds on the expected run time of an EA on HASMAJORITY . We note that this framework is also useful in the setting of deletion-robust optimization, as studied by Bian et al. [3], since MAJORITY occurs as a subproblem there.

We complement our theoretical analyses with empirical investigations of the RLS variant RLS_ℓ , which flips exactly $\ell \in [1, n] \cap \mathbb{N}$ bits (chosen uniformly among all ℓ -cardinality subsets), on MAJORITY . For $r = \lfloor \sqrt{n} \rfloor$, we consider the empirical run time for different values of ℓ , and we find that a value around $n/2$ leads to the lowest run time (Figure 5), which is by orders of magnitude lower than the empirical run time of conventional RLS. Additional results, which look into the trajectory of the best solution so far over time, reveal that while instantiations of RLS_ℓ for different values of ℓ have qualitatively the same random-walk behavior, the larger step size for larger values of ℓ makes it easier to find a global optimum (Figure 6). However, we note that there is a limit to how many bits to flip during a single iteration makes sense, as, for example, inverting the bit string in each iteration makes RLS_n alternate between just two solutions. Nonetheless, a high mutation rate in the order of n seems beneficial for our setting.

b) *Related work:* Analyzing the performance of EAs on plateaus by theoretical means is a well-established concept. The arguably most studied benchmark function with a plateau is JUMP , existing in different variants, many of which were

¹Note that, in contrast to Figure 1, we denote the dimension of HASMAJORITY_r (defined over a block) now by n and not anymore by k .

proposed recently [5], [6], [7], [8]. However, the problem posed by the plateau in JUMP is typically different than from crossing a plateau, as we discuss in the following. In its classical variant, JUMP consists of a slope toward the unique global optimum, which itself is surrounded by solutions with worse function values (called the *valley*). The plateau is the set of points with second-best function value. For JUMP, the behavior of an EA on the plateau is typically not that important, but rather that the valley is overcome, oftentimes referred to as *jumping* over the valley. This is in contrast to crossing a plateau, where the behavior of an EA on the plateau is very important. The main importance of jumping remains true for JUMP when shifting the global optimum [7] and/or increasing the number of global optima to jump to [6], [8]. However, the picture changes when considering EAs that apply crossover, that is, combining different solutions when creating new ones. Then, the random walk on the plateau becomes relevant for the expected run time on JUMP [9], [10], [11]. Nonetheless, in such a setting, it is primarily important to create diverse solutions on the plateau, in order to leave it via crossover, instead of crossing the plateau (by finding a better solution next to it).

One variant of JUMP, recently introduced by Antipov and Doerr [5], which removes the valley and increases the plateau, actually requires dealing with crossing the plateau. The authors analyze the $(1+1)$ EA on this variant and determine an exact run time bound (up to lower-order terms) via arguments on Markov chains. In contrast to our setting, where the set of global optima is typically very large (in the order of $\binom{n}{n/2+r}$), their function has a unique global optimum.

In another recent paper, Bian et al. [3] analyzed the $(1+1)$ EA on MAJORITY as a side result when considering deletion-robust linear optimization. Using our notion of MAJORITY, their result holds for values of $r = O(\sqrt{n \log n})$, that is, not for the entire range of r up to $n/2$. Further, their result does not seem to be tight, as for values of $r = \Theta(1)$, their bound is $O(n^2)$, whereas we show a constant bound, albeit for RLS.

c) Outline: In Section II, we introduce our notation, formalize our problem, and provide the mathematical tools we use for our analysis. Especially, we prove Theorems 4 and 5, which we believe are of independent interest. In Section III, we analyze RLS on HASMAJORITY (Theorem 6) and MAJORITY (Theorem 7). Further, we provide the theorem for decomposing stopping times (Theorem 8), show how it can be used to relate the expected run times of EAs on HASMAJORITY and MAJORITY (Corollary 9), and put everything together to prove our main result, the run time bound for RLS optimizing ONEMAX with added neutrality from the W-model (Corollary 13). Section IV describes our empirical results. Last, we conclude with Section V and suggest ideas for future work.

II. PRELIMINARIES

Let \mathbb{N} denote the set of natural numbers, including 0. For all $m, n \in \mathbb{N}$, let $[m..n] := [m, n] \cap \mathbb{N}$, and let $[m] := [1..m]$. Further, let \mathbb{R} denote the set of reals, and let $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. For the sake of conciser notations, we define that $\infty \cdot 0 = 0 = -\infty \cdot 0$. We say that a random

Algorithm 1: Random local search (RLS) with initialization distribution D , maximizing $f: \{0, 1\}^n \rightarrow \mathbb{R}$.

```

1  $\mathbf{x}^{(0)} \sim D$ ;
2 for  $t \in \mathbb{N}$  do
3    $\mathbf{y} \leftarrow$  flip exactly one bit in  $\mathbf{x}^{(t)}$ , chosen uniformly
   at random;
4   if  $f(\mathbf{y}) \geq f(\mathbf{x}^{(t)})$  then  $\mathbf{x}^{(t+1)} \leftarrow \mathbf{y}$ ;
5   else  $\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)}$ ;

```

variable X over \mathbb{R} is *integrable* if and only if $E[|X|] < \infty$. Note that integrability of X implies that X is almost surely finite. We extend this concept to random processes and say that $(X_t)_{t \in \mathbb{N}}$ is integrable if and only if for each $t \in \mathbb{N}$, it holds that X_t is integrable.

Let $n \in \mathbb{N}_{>0}$. We call each $\mathbf{x} \in \{0, 1\}^n$ an *individual* and its components *bits*. Further, for all $b \in \{0, 1\}$, let $|\mathbf{x}|_b := |\{i \in [n] \mid x_i = b\}|$. We call any real-valued function over individuals a *fitness function*.

Let $n \in \mathbb{N}_{>0}$ be even, and let $r \in [0..n/2]$. We consider the two fitness functions HASMAJORITY $_r$ and MAJORITY $_r$, where, for all $\mathbf{x} \in \{0, 1\}^n$, it holds that

$$\begin{aligned} \text{HASMAJORITY}_r(\mathbf{x}) &= \mathbb{1}(\max\{|\mathbf{x}|_0, |\mathbf{x}|_1\} \geq n/2 + r) \text{ and} \\ \text{MAJORITY}_r(\mathbf{x}) &= \mathbb{1}(|\mathbf{x}|_1 \geq n/2 + r), \end{aligned}$$

where $\mathbb{1}(\mathcal{E})$ is the indicator function that evaluates to 1 if event \mathcal{E} occurs and that evaluates to 0 otherwise. When we drop the index r and only mention HASMAJORITY or MAJORITY, we mean that the respective statement holds for all $r \in [0..n/2]$.

We note that HASMAJORITY and MAJORITY are special cases of the functions mentioned in Section I with the target string $z \in \{0, 1\}^n$ being the all-1s string. Since we only consider algorithms that treat 0s and 1s symmetrically, analyzing HASMAJORITY and MAJORITY as defined above covers without loss of generality the more general function class discussed in Section I.

We consider *evolutionary algorithms* (EAs) to be (possibly random) sequences $(\mathbf{x}^{(t)})_{t \in \mathbb{N}}$ over $\{0, 1\}^n$. Typically, the sequence of an EA A depends (among others) on a fitness function f , and is not well-defined otherwise. We refer to well-defined sequences by saying that A *optimizes* f .

A specific EA that we are interested in and that optimizes a fitness function is *random local search* (RLS, Algorithm 1). Note that for RLS to optimize f and the respective sequence to be well-defined, in addition to f , an initialization distribution needs to be specified. Typically, the uniform distribution is chosen. However, since we require for some of our results that RLS starts at a specific position in the search space, we consider general initialization distributions.

Given an EA $(\mathbf{x}^{(t)})_{t \in \mathbb{N}}$ optimizing a fitness function f , we call $\inf\{t \in \mathbb{N} \mid f(\mathbf{x}^{(t)}) = \max \text{rng}(f)\}$ the *run time* of the algorithm on f . Note that the set that the infimum is taken over may be empty. To this end, we define $\inf \emptyset = \infty$. Further note that the run time can be a random variable. We refer to the expectation of the run time as the *expected run time*.

A. Tools

Besides basic tools from probability theory, we apply drift analysis [12]. Most prominently, we use the additive-drift theorem, originally introduced by He and Yao [13], as well as the multiplicative-drift theorem, originally introduced by Doerr et al. [14]. We state the theorems in a general fashion, as provided by Kötzing and Krejca [15], removing some unnecessary assumptions.

Theorem 1 (Additive drift, upper bound [15, Theorem 7]). *Let $(X_t)_{t \in \mathbb{N}}$ be an integrable random processes over $\mathbb{R}_{\geq 0}$, adapted to a filtration $(\mathcal{F}_t)_{t \in \mathbb{N}}$, and let T be a stopping time with respect to \mathcal{F} . Assume that there is a $\delta \in \mathbb{R}_{>0}$ such that, for all $t \in \mathbb{N}$, it holds that*

$$(X_t - \mathbb{E}[X_{t+1} \mid \mathcal{F}_t]) \cdot \mathbf{1}(t < T) \geq \delta \cdot \mathbf{1}(t < T). \quad (1)$$

Then $\mathbb{E}[T \mid \mathcal{F}_0] \leq (X_0 - \mathbb{E}[X_T \mid \mathcal{F}_0])/\delta$.

Theorem 2 (Multiplicative drift, upper bound [15, Corollary 16]). *Let $(X_t)_{t \in \mathbb{N}}$ be an integrable random processes over \mathbb{R} , adapted to a filtration $(\mathcal{F}_t)_{t \in \mathbb{N}}$, with $X_0 \geq 1$, and let $T = \inf\{t \in \mathbb{N} \mid X_t < 1\}$. Assume that there is a $\delta \in \mathbb{R}_{>0}$ such that, for all $t \in \mathbb{N}$, it holds that*

$$(X_t - \mathbb{E}[X_{t+1} \mid \mathcal{F}_t]) \cdot \mathbf{1}(t < T) \geq \delta \cdot X_t \cdot \mathbf{1}(t < T). \quad (2)$$

Then $\mathbb{E}[T \mid \mathcal{F}_0] \leq (1 + \ln(X_0))/\delta$.

In addition to these theorems, we apply a general version of Wald's equation (Theorem 5), which we prove using a new general drift theorem (Theorem 4), as we could not find a statement this general in the literature. We prove Theorem 4 by using the optional-stopping theorem (Theorem 3), similar to how Kötzing and Krejca [15] derived their results.

Theorem 3 (Optional stopping [16, Theorems 4.8.4 and 4.8.5], [17, Slide 12]). *Let S be a stopping time with respect to a filtration $(\mathcal{F}_i)_{i \in \mathbb{N}}$, and let $(X_i)_{i \in \mathbb{N}}$ be a random process over \mathbb{R} , adapted to \mathcal{F} .*

- If X is a non-negative supermartingale, then $\mathbb{E}[X_S \mid \mathcal{F}_0] \leq X_0$.
- If $\mathbb{E}[S] < \infty$ and if X is a submartingale such that there is a $c \in \mathbb{R}$ such that, for all $i \in \mathbb{N}$, it holds that $\mathbb{E}[(X_i - X_{i+1}) \mid \mathcal{F}_i] \cdot \mathbf{1}(i < S) \leq c$, then $\mathbb{E}[X_S \mid \mathcal{F}_0] \geq X_0$.

The basic-drift theorem below is a generalization of the most commonly used drift theorems. However, in contrast to those, it relates the accumulated drift to the overall change in potential instead of explicitly solving for the first-hitting time. It is thus more akin to the (deterministic) potential method [18, Chapter 17.3], extending it to random variables. This allows to estimate the accumulated *cost* of operations rather than only the *amount* of operations.

Theorem 4 (Basic drift). *Let S be a stopping time with respect to a filtration $(\mathcal{F}_i)_{i \in \mathbb{N}}$, and let $(X_i)_{i \in \mathbb{N}}$ and $(\delta_i)_{i \in \mathbb{N}}$ be random processes over \mathbb{R} such that $(X_i \cdot \mathbf{1}(i \leq S))_{i \in \mathbb{N}}$ and $(\delta_i \cdot \mathbf{1}(i < S))_{i \in \mathbb{N}}$ are integrable and adapted to \mathcal{F} .*

- Assume that $(X_i \cdot \mathbf{1}(i \leq S))_{i \in \mathbb{N}}$ and $(\delta_i \cdot \mathbf{1}(i < S))_{i \in \mathbb{N}}$ are non-negative and that for all $i \in \mathbb{N}$, it holds that

$$(X_i - \mathbb{E}[X_{i+1} \mid \mathcal{F}_i]) \cdot \mathbf{1}(i < S) \geq \delta_i \cdot \mathbf{1}(i < S). \quad (\text{DC-u})$$

Then

$$\mathbb{E}\left[\sum_{i \in [S]} \delta_{i-1} \mid \mathcal{F}_0\right] \leq X_0 - \mathbb{E}[X_S \mid \mathcal{F}_0]. \quad (3)$$

- Assume that $\mathbb{E}[S] < \infty$, and that there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}$, it holds that $\mathbb{E}[|X_i - X_{i+1} - \delta_i| \mid \mathcal{F}_i] \cdot \mathbf{1}(i < S) \leq c$, and that for all $i \in \mathbb{N}$, it holds that

$$(X_i - \mathbb{E}[X_{i+1} \mid \mathcal{F}_i]) \cdot \mathbf{1}(i < S) \leq \delta_i \cdot \mathbf{1}(i < S). \quad (\text{DC-l})$$

Then

$$\mathbb{E}\left[\sum_{i \in [S]} \delta_{i-1} \mid \mathcal{F}_0\right] \geq X_0 - \mathbb{E}[X_S \mid \mathcal{F}_0].$$

Proof. Let $(Z_i)_{i \in \mathbb{N}}$ be such that for all $i \in \mathbb{N}$, it holds that $Z_i = X_i + \sum_{k \in [i]} \delta_{k-1}$. We aim to apply Theorem 3 to the stopped process $(Y_i)_{i \in \mathbb{N}}$ of Z , and to S and \mathcal{F} . To this end, for all $i \in \mathbb{N}$, let

$$Y_i = Z_i \cdot \mathbf{1}(i \leq S) + Z_S \cdot \mathbf{1}(i > S).$$

Note that, all by assumption, S is a stopping time with respect to \mathcal{F} , and Y is integrable and adapted to \mathcal{F} , due to $(X_i \cdot \mathbf{1}(i \leq S))_{i \in \mathbb{N}}$ and $(\delta_i \cdot \mathbf{1}(i < S))_{i \in \mathbb{N}}$ having these properties.

We are left to show that Y is a super- or a submartingale with additional properties. To this end, we first consider the expected change of Y . Let $i \in \mathbb{N}$. Noting that S is integer, observe that

$$\begin{aligned} Y_i - \mathbb{E}[Y_{i+1} \mid \mathcal{F}_i] &= \mathbf{1}(i=S) + \mathbf{1}(i < S) \\ &= Z_i \cdot \mathbf{1}(i \leq S) + Z_S \cdot \mathbf{1}(i > S) \\ &\quad - \mathbb{E}[Z_{i+1} \cdot \mathbf{1}(i+1 \leq S) + Z_S \cdot \mathbf{1}(i+1 > S) \mid \mathcal{F}_i] \\ &= (Z_i - \mathbb{E}[Z_{i+1} \mid \mathcal{F}_i]) \cdot \mathbf{1}(i < S) + Z_i \cdot \mathbf{1}(i = S) \\ &\quad + Z_S \cdot \mathbb{E}[\mathbf{1}(i > S) - \mathbf{1}(i+1 > S) \mid \mathcal{F}_i] \\ &= (Z_i - \mathbb{E}[Z_{i+1} \mid \mathcal{F}_i]) \cdot \mathbf{1}(i < S) \\ &= (X_i - \mathbb{E}[X_{i+1} \mid \mathcal{F}_i] - \delta_i) \cdot \mathbf{1}(i < S). \end{aligned} \quad (4)$$

Whether Y is a super- or a submartingale is determined by the sign of equation (4), noting that $\mathbf{1}(i < S) \geq 0$. To this end, we consider the two cases of Theorem 4 separately.

Case a). By inequality (DC-u), it follows that equation (4) is greater or equal to zero. Thus, Y is a supermartingale. Further, due to the assumptions of Theorem 4 case a), Y is non-negative. Applying Theorem 3 case a), by the linearity of expectation and noting that $S \geq 0$, we get

$$\begin{aligned} \mathbb{E}[X_S \mid \mathcal{F}_0] + \mathbb{E}\left[\sum_{i \in [S]} \delta_{i-1} \mid \mathcal{F}_0\right] &= \mathbb{E}[Y_S \mid \mathcal{F}_0] \\ &\leq Y_0 = X_0. \end{aligned} \quad (5)$$

Subtracting by $\mathbb{E}[X_S \mid \mathcal{F}_0]$ proves this case.

Case b). Similar to the previous case, by inequality (DC-l),

it follows that [equation \(4\)](#) is less or equal to zero. Thus, Y is a submartingale. By the assumptions of [Theorem 4 case b\)](#), it follows that Y has uniformly bounded expected differences. Recalling that, by assumption, $E[S] < \infty$ in this case and by applying [Theorem 3 case b\)](#), the same result as in [equation \(5\)](#) follows but with the inequality sign flipped, concluding this case and thus the proof. \square

An application of [Theorem 4](#) is the generalization of Wald's equation below. In contrast to more common statements of Wald's equation, this version neither assumes that the individual summands follow the same law nor that they are independent. In the context of run time analysis, it is useful when aiming to split the run time into phases. Instead of having to calculate the entire expected run time all at once, the theorem states that it is sufficient to calculate the expected run time of each phase (and *then* add them). The latter is typically far easier, especially since the theorem allows to choose a filtration. This result was recently applied in the analysis of infection processes, where it was used in order to translate statements about discrete time to continuous time [[19](#)].

Theorem 5 (Generalized Wald's equation). *Let S be an integrable stopping time with respect to a filtration $(\mathcal{F}_i)_{i \in \mathbb{N}}$. Further, let $(X_i)_{i \in \mathbb{N}_{>0}}$ be a random process over $\mathbb{R}_{\geq 0}$ such that $\sum_{i \in [S]} X_i$ is integrable and such that there exists a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}$, it holds that $E[|X_{i+1} - E[X_{i+1} | \mathcal{F}_i]| | \mathcal{F}_i] \cdot \mathbf{1}(i < S) \leq c$. Then $E[\sum_{i \in [S]} X_i | \mathcal{F}_0] = E[\sum_{i \in [S]} E[X_i | \mathcal{F}_{i-1}] | \mathcal{F}_0]$.*

Checking that $\sum_{i \in [S]} X_i$ is integrable can prove challenging in certain cases. However, if this sum can be expressed as a single stopping time, [Theorem 4 case a\)](#) might be used to show that the sum is integrable.

Proof of [Theorem 5](#). Let $(Y_i)_{i \in \mathbb{N}}$ be such that for all $i \in \mathbb{N}$, it holds that $Y_i = E[\sum_{j=i+1}^S X_j | \mathcal{F}_i]$. Further, let $(\delta_i)_{i \in \mathbb{N}}$ be such that for all $i \in \mathbb{N}$, it holds that $\delta_i = E[X_{i+1} | \mathcal{F}_i]$. Note that $(Y_i \cdot \mathbf{1}(i \leq S))_{i \in \mathbb{N}}$ and $(\delta_i \cdot \mathbf{1}(i < S))_{i \in \mathbb{N}}$ are adapted to \mathcal{F} due to the conditional expectation and that they are integrable since $\sum_{i \in [S]} X_i$ is integrable and X is non-negative.

We aim to apply both cases of [Theorem 4](#) to Y and δ with S and \mathcal{F} . To this end, we note that for all $i \in \mathbb{N}$, by the tower rule and the linearity of the conditional expectation, we have

$$\begin{aligned} & (Y_i - E[Y_{i+1} | \mathcal{F}_i]) \cdot \mathbf{1}(i < S) \\ &= \left(E \left[\sum_{j=i+1}^S X_j \mid \mathcal{F}_i \right] \right. \\ & \quad \left. - E \left[E \left[\sum_{j=i+2}^S X_j \mid \mathcal{F}_{i+1} \right] \mid \mathcal{F}_i \right] \right) \cdot \mathbf{1}(i < S) \\ &= \left(E \left[\sum_{j=i+1}^S X_j \mid \mathcal{F}_i \right] \right. \\ & \quad \left. - E \left[\sum_{j=i+2}^S X_j \mid \mathcal{F}_i \right] \right) \cdot \mathbf{1}(i < S) \\ &= E[X_{i+1} | \mathcal{F}_i] \cdot \mathbf{1}(i < S) = \delta_i \cdot \mathbf{1}(i < S), \end{aligned}$$

which shows that [inequalities \(DC-1\)](#) and [\(DC-u\)](#) are satisfied. We now apply both cases of [Theorem 4](#) separately in order to show both directions of the equality of [Theorem 5](#).

Case 1: Note that $(Y_i \cdot \mathbf{1}(i \leq S))_{i \in \mathbb{N}}$ is non-negative because X is. By [Theorem 4 case a\)](#),

$$\begin{aligned} E \left[\sum_{i \in [S]} E[X_i | \mathcal{F}_{i-1}] \mid \mathcal{F}_0 \right] &= E \left[\sum_{i \in [S]} \delta_{i-1} \mid \mathcal{F}_0 \right] \\ &\leq Y_0 - E[Y_S | \mathcal{F}_0] = E \left[\sum_{i \in [S]} X_i \mid \mathcal{F}_0 \right]. \end{aligned}$$

Case 2: By assumption, S is integrable, and there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}$, it holds that

$$\begin{aligned} & E[|Y_i - Y_{i+1} - \delta_i| | \mathcal{F}_i] \cdot \mathbf{1}(i < S) \\ &= E \left[\left| \sum_{j=i+1}^S X_j - \sum_{j=i+2}^S X_j - E[X_{i+1} | \mathcal{F}_i] \right| \mid \mathcal{F}_i \right] \\ & \quad \cdot \mathbf{1}(i < S) \\ &= E[|X_{i+1} - E[X_{i+1} | \mathcal{F}_i]| | \mathcal{F}_i] \cdot \mathbf{1}(i < S) \leq c. \end{aligned}$$

By [Theorem 4 case b\)](#) and the same calculations as at the end of the previous case but with an inverted inequality sign, we get a matching lower bound and conclude the proof. \square

III. THEORETICAL RESULTS FOR RLS

We analyze the expected run time of RLS on HASMAJORITY ([Theorem 6](#)) and on MAJORITY ([Theorem 7](#)). We exploit the similarities between both functions, deriving the result for MAJORITY based on the result for HASMAJORITY. In fact, we prove a very general statement ([Corollary 9](#)), which shows how to derive the expected run time for a great range of algorithms optimizing MAJORITY, given their expected run time on HASMAJORITY and some additional, related expected run times. This result is a special case of an even more general theorem ([Theorem 8](#)), which decomposes an expected stopping time into smaller intervals, which are easier to analyze. Thus, overall, deriving a good bound on the expected run time on HASMAJORITY is essential. Last, we show how our result on MAJORITY implies run time bounds for the classical ONEMAX when transformed by neutrality of the W-model ([Corollary 13](#)).

When RLS optimizes HASMAJORITY, before finding an optimal solution for the first time, it performs a random walk, as it always replaces the solution from the previous iteration with the new solution. This random walk is biased toward solutions with as many 0s as 1s (the *center*), since, in order to create a new solution, RLS flips one bit uniformly at random from the solution from the previous iteration and any imbalance in the number of 0s and 1s favors flipping those bits that occur in a higher quantity. Thus, the number of possible ways of an individual to get to the center compared to those toward an individual with all 1s or all 0s grows exponentially in the distance to the center. Since the optima of HASMAJORITY are off-center, this results in the expected run time growing exponentially with respect to a basis that we call $\lambda \in \mathbb{R}_{>1}$, which is *not* necessarily bounded away from 1 by a constant.

Let $n \in \mathbb{N}_{>0}$ be even, and let $r \in [n/2]$. The expected run time of RLS on HASMAJORITY $_r$ majorly depends on

$$\lambda = \frac{3r(n + 2(r - 1))}{3r(n - 2(r - 1)) - 2n} = 1 + \frac{2(n + 6r(r - 1))}{3r(n - 2(r - 1)) - 2n}, \quad (6)$$

which is greater than 1. We discuss this term and its impact on the expected run time of RLS after stating our main results.

Theorem 6. Let $n \in \mathbb{N}_{>0}$ be even, $r \in [n/2]$, let λ be as in equation (6), and let $h: \{0, 1\}^n \rightarrow \mathbb{R}_{\geq 0}$ with

$$\mathbf{x} \mapsto \begin{cases} \lambda^r - \lambda^{\max\{|\mathbf{x}|_0, |\mathbf{x}|_1\} - n/2} & : \max\{|\mathbf{x}|_0, |\mathbf{x}|_1\} < \frac{n}{2} + r, \\ 0 & : \text{otherwise.} \end{cases}$$

Let $\mathbf{x}^{(0)}$ denote the (possibly random) initial individual of RLS. The expected run time of RLS on HASMAJORITY $_r$, conditional on $\mathbf{x}^{(0)}$, is at most $3r \cdot h(\mathbf{x}^{(0)})/(\lambda - 1)$.

Theorem 7. Let $n \in \mathbb{N}_{>0}$ be even, $r \in [n/2]$, and let λ be as in equation (6). For sufficiently large n , the expected run time of RLS on MAJORITY $_r$ with uniform initialization is at most $6r \cdot (\lambda^r - 1)/(\lambda - 1) + n(1 + \ln(r))/2$.

Theorem 6 shows that there is a *drastic* change in the expected run time of RLS on HASMAJORITY with respect to r . In order to see this, recall equation (6) and note that $h(\mathbf{x}^{(0)}) \leq \lambda^r$. If $r = \Theta(1)$, then $\lambda = 1 + \Theta(1)$, and the expected run time is constant. If $r = \omega(1) \cap O(\sqrt{n})$, then $\lambda = 1 + O(1/r)$, and the expected run time is $O(r^2)$, as $\lambda^r \leq e^{O(1)}$. Especially, if $r = O(\sqrt{n})$, then the expected run time is at most linear in n . If $r = \omega(\sqrt{n}) \cap o(n)$, then $\lambda = 1 + O(r/n)$, and the expected run time is $ne^{O(r^2/n)}$, which is superlinear in n and even superpolynomial in n for $r = \omega(\sqrt{n \log n})$. Last, if there is a $c = \Theta(1)$ with $c < 1/2$ and an $s \in \mathbb{N}$ with $s \leq cn$ such that $r = n/2 - s$, then $\lambda = 1 + O(n/(s + 1))$, and the expected run time is $(s + 1)e^{O(n^2/(s + 1))}$, which is at least exponential in n .

This drastic increase in the expected run time carries over to Theorem 7, whose bound we compare to an empirically determined one, as depicted in Figure 3 for $n = 10000$. Due to the steep growth of the run time in r , we only get empirical results up to $r = 170 = 1.7\sqrt{n}$. Nonetheless, the empirical expected run time suggests an exponential growth in r . However, the growth does not seem to be as drastic as the theoretical upper bound of Theorem 7 suggests, which is larger than the empirical average by more than a constant factor.

A. Run Time Results on HASMAJORITY

In our proof of Theorem 6, we aim to apply the additive-drift theorem (Theorem 1). To this end, letting \mathbf{x} denote the currently best individual of RLS in each iteration, we consider the distance $n/2 + r - \max\{|\mathbf{x}|_0, |\mathbf{x}|_1\}$ of the majority of bits of \mathbf{x} to the optimum value $n/2 + r$ of HASMAJORITY $_r$. Since it becomes less likely to decrease this distance the smaller it is, we choose a potential function that scales the space exponentially, counteracting this decline in probability.

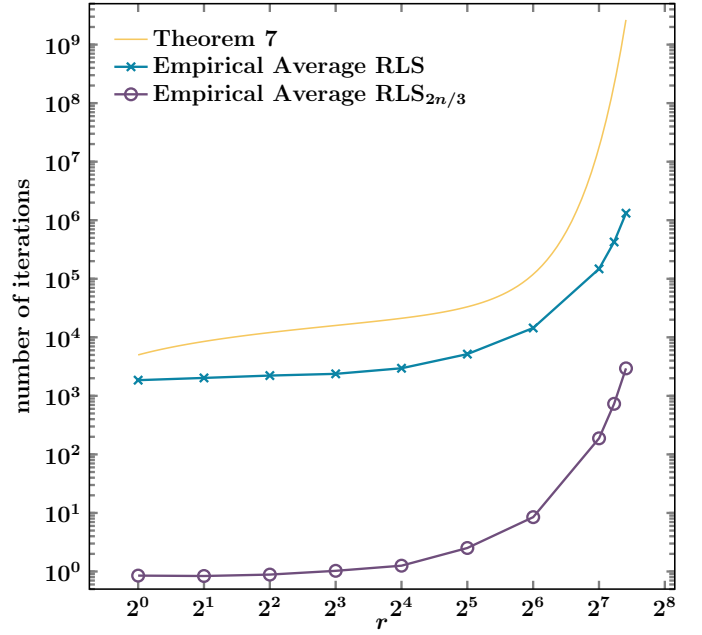


Fig. 3: The theoretical (top, orange curve) and empirical expected run time of RLS (center, blue curve with \times) and RLS $_{2n/3}$ (bottom, purple curve with \circ) on MAJORITY $_r$ for $n = 10000$ and $r \in \{2^i \mid i \in \{0\} \cup [7]\} \cup \{150, 170\}$. The theoretical run time is the upper bound from Theorem 7. For the empirical run times, for each value of r stated above, the average of 1000 independent runs of RLS and of RLS $_{2n/3}$ with uniform initialization is depicted. For more information regarding RLS $_{2n/3}$, please see Section IV-A; the value $2n/3$ is near-optimal, as shown in Figure 5, resulting in a drastically improved run time.

Proof of Theorem 6. For all $t \in \mathbb{N}$, let $X_t = \max\{|\mathbf{x}^{(t)}|_0, |\mathbf{x}^{(t)}|_1\}$, and let $(\mathcal{F}_t)_{t \in \mathbb{N}}$ denote the natural filtration of X . In addition, let $g: [n/2..n] \rightarrow \mathbb{R}_{\geq 0}$ with

$$m \mapsto \begin{cases} \lambda^r - \lambda^{m - n/2} & : m < n/2 + r, \\ 0 & : \text{otherwise.} \end{cases}$$

Last, let $T = \inf\{t \in \mathbb{N} \mid g(X_t) = 0\}$, and note that T is a stopping time with respect to \mathcal{F} and that it denotes the first point in time such that RLS found an optimum of HASMAJORITY $_r$, as $X_T \geq n/2 + r$. We show that $g(X)$ satisfies inequality (1) with \mathcal{F} and T . Theorem 1 then yields the desired bound.

Let $t \in \mathbb{N}$, and assume that $t < T$, as inequality (1) is trivially satisfied otherwise. Note that X_t increases by 1 if a bit that is not the strict majority in $\mathbf{x}^{(t)}$ is chosen to be flipped, which happens with probability $(n - X_t)/n$ if $X_t > n/2$, and with probability 1 if $X_t = n/2$. With the remaining probability, X_t decreases by 1.

For $X_t = n/2$, we get

$$g(X_t) - \mathbb{E}[g(X_{t+1}) \mid \mathcal{F}_t] = \lambda^r - 1 - (\lambda^r - \lambda) = \lambda - 1. \quad (7)$$

For the remaining case $X_t \in [n/2 + 1..n/2 + r - 1]$, we get

$$g(X_t) - \mathbb{E}[g(X_{t+1}) \mid \mathcal{F}_t]$$

$$\begin{aligned}
 &= \lambda^r - \lambda^{X_t - n/2} - \left(\lambda^r - \lambda^{X_t - 1 - n/2} \right) \frac{X_t}{n} \\
 &\quad - \left(\lambda^r - \lambda^{X_t + 1 - n/2} \right) \frac{n - X_t}{n} \\
 &= \lambda^{X_t - n/2} \underbrace{\left(\frac{1}{\lambda} \cdot \frac{X_t}{n} + \lambda \cdot \frac{n - X_t}{n} - 1 \right)}_{=:A}. \tag{8}
 \end{aligned}$$

Let $b = 2(r - 1)$, $a = 3r(n - b)(n + b)/(3r(n - b) - 2n)$, and note that $\lambda = a/(n - b)$. Since λ is greater than 1, it follows that A is decreasing in X_t and thus minimal for $X_t = n/2 + r - 1 = (n + b)/2$. Substituting $\lambda = a/(n - b)$, this yields

$$\begin{aligned}
 A &\geq \frac{n - b}{a} \cdot \frac{n + b}{2n} + \frac{a}{n - b} \cdot \frac{n - b}{2n} \\
 &= \frac{(n - b)(n + b) + a^2}{2an} = \underbrace{\frac{n^2 - b^2}{2an}}_{=:B} + \frac{a}{2n}. \tag{9}
 \end{aligned}$$

Noting that $a = 3r(n^2 - b^2)/(3r(n - b) - 2n)$, we get

$$B = \frac{3r(n - b) - 2n}{2n \cdot 3r} = \frac{n - b}{2n} - \frac{1}{3r}.$$

Substituting this back into [inequality \(9\)](#) yields

$$A \geq \underbrace{\frac{a + n - b}{2n}}_{=:C} - \frac{1}{3r}. \tag{10}$$

Again, by substituting $a = 3r(n - b)(n + b)/(3r(n - b) - 2n)$, we get

$$\begin{aligned}
 C &= \frac{1}{2n} \left(\frac{3r(n - b)(n + b)}{3r(n - b) - 2n} + \frac{(n - b)(3r(n - b) - 2n)}{3r(n - b) - 2n} \right) \\
 &= \frac{n - b}{2n} \cdot \frac{3r(n + b + n - b) - 2n}{3r(n - b) - 2n} = \frac{3r(n - b) - (n - b)}{3r(n - b) - 2n} \\
 &= 1 + \frac{n + b}{3r(n - b) - 2n} = 1 + \frac{1}{3r} \cdot \frac{3r(n + b)}{3r(n - b) - 2n} \\
 &= 1 + \frac{a}{n - b} \cdot \frac{1}{3r}.
 \end{aligned}$$

Substituting this back into [inequality \(10\)](#) and recalling that $\lambda = a/(n - b)$ yields $A \geq 1 + (\lambda - 1)/(3r)$. Substituting this bound into [equation \(8\)](#) ultimately yields

$$g(X_t) - \mathbb{E}[g(X_{t+1}) \mid \mathcal{F}_t] \geq \lambda^{X_t - n/2} \cdot (\lambda - 1) \cdot \frac{1}{3r}. \tag{11}$$

Applying [Theorem 1](#) with $\delta = (\lambda - 1)/(3r)$ being the minimum of [inequalities \(7\)](#) and [\(11\)](#) and noting that $\mathbb{E}[g(X_T) \mid \mathcal{F}_0] = 0$ concludes the proof. \square

B. Run Time Results on MAJORITY

We show how to derive a run time bound of an EA A for MAJORITY based on a run time bound for HASMAJORITY ([Corollary 9](#)). The argument is based on the observation that whenever A finds an optimum of HASMAJORITY, this solution can also be an optimum of MAJORITY. If this is *not* the case, that is, if the current solution has a majority of 0s, we wait until A gets back to a solution of at least $n/2$ 1s. Then, we wait again until it finds an optimum of HASMAJORITY, repeating this argument until A finds an optimum of MAJORITY.

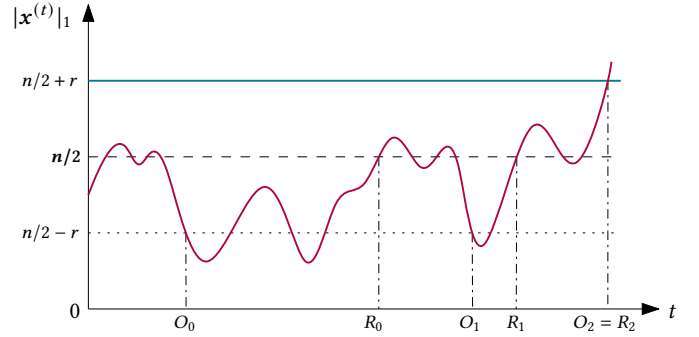


Fig. 4: A visualization of the definitions of the random variables $(O_i)_{i \in \mathbb{N}}$ and $(R_i)_{i \in \mathbb{N}}$ as defined in [equation \(12\)](#), needed for [Corollary 9](#). The red wavy line depicts an example of how the number of 1s in the best solution of an EA (given by the sequence $(\mathbf{x}^{(t)})_{t \in \mathbb{N}}$) changes over time (t). For the sake of readability, we depict this line as continuous although it is discrete and can have jumps. The stopping times $(O_i)_{i \in \mathbb{N}}$ and $(R_i)_{i \in \mathbb{N}}$ always alternate, by definition.

More formally, we define the following two interleaving sequences of stopping times, which we also visualize in [Figure 4](#):

$$O_0 = \inf\{t \in \mathbb{N} \mid \text{HASMAJORITY}_r(\mathbf{x}^{(t)}) = 1\}, \tag{12}$$

as well as, for all $i \in \mathbb{N}$,

$$R_i = \inf\{t \in \mathbb{N} \mid t \geq O_i \wedge |\mathbf{x}^{(t)}|_1 \geq n/2\}, \text{ and}$$

$$O_{i+1} = \inf\{t \in \mathbb{N} \mid t > R_i \wedge \text{HASMAJORITY}_r(\mathbf{x}^{(t)}) = 1\}.$$

The sequence O denotes the points in time (interleaved by R) where A can have found an optimum of MAJORITY, as it found an optimum of HASMAJORITY. In the context of proving a run time bound for MAJORITY, O denotes the points in time from where we wait until A returns to an individual with at least as many 1s as 0s in order to repeat our argument. The sequence R denotes those points in time where we repeat our argument. [Theorem 5](#) formalizes this argument and yields a bound on the overall expected run time on MAJORITY.

Since the idea of this argument holds for any interleaved assortment of stopping times, we first prove a more general theorem ([Theorem 8](#)) before we state the result specific to our setting ([Corollary 9](#)). This theorem generalizes the idea of the stopping times from [equation \(12\)](#), but it does not specify their interpretation.

Theorem 8 (Decomposition argument). *Let $(\mathcal{F}_i)_{i \in \mathbb{N}}$ be a filtration, let S be a stopping time with respect to \mathcal{F} , and let $(O_i)_{i \in \mathbb{N}}$ and $(R_i)_{i \in \mathbb{N}}$ be stopping times with respect to \mathcal{F} such that for all $i \in \mathbb{N}$, it holds that $O_i \leq R_i < O_{i+1}$. Moreover, let $U = \sup\{i \in \mathbb{N} \mid O_i \leq S\}$, and assume that there is a $p_0 \in (0, 1]$ such that $\Pr[O_0 = S] = p_0$.*

Assume that $\sum_{i \in [U]} (O_i - O_{i-1})$ and U are integrable as well as that there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}_{>0}$, it holds that $\mathbb{E}[O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]] \mid \mathcal{F}_{R_{i-1}}] \leq c$ and $\mathbb{E}[R_{i-1} - \mathbb{E}[R_{i-1} \mid \mathcal{F}_{O_{i-1}}]] \mid \mathcal{F}_{O_{i-1}}] \leq c$.

Last, let L denote the event that $U \geq 1$. Then

$$\begin{aligned} \mathbb{E}[S] = \mathbb{E}[O_0] + (1 - p_0) \mathbb{E} \left[\sum_{i \in [U]} \left(\mathbb{E}[O_i - R_{i-1} \mid \mathcal{F}_{R_{i-1}}] \right. \right. \\ \left. \left. + \mathbb{E}[R_{i-1} - O_{i-1} \mid \mathcal{F}_{O_{i-1}}] \right) \mid L \right], \end{aligned}$$

where we interpret the second summand to be 0 if $1 - p_0 = 0$.

Proof. We aim to express S as a sum of time intervals that reflect the repeats as outlined at the beginning of Section III-B. To this end, for all $i \in \mathbb{N}_{>0}$, let $\Delta_i = O_i - O_{i-1}$. Note that this implies that $S = T + \sum_{i \in [U]} \Delta_i$, since $T = O_0$, $S = O_U$, and the sum is telescoping.

By the linearity of expectation, it follows that $\mathbb{E}[S] = \mathbb{E}[T] + \mathbb{E}[\sum_{i \in [U]} \Delta_i]$. Note that $\mathbb{E}[\sum_{i \in [U]} \Delta_i \mid \bar{L}] = 0$ as well as $\Pr[L] = 1 - p_0$. Thus, $\mathbb{E}[\sum_{i \in [U]} \Delta_i] = (1 - p_0) \mathbb{E}[\sum_{i \in [U]} \Delta_i \mid L]$, which is 0 if $\Pr[L] = 0$.

Let $i \in \mathbb{N}_{>0}$, and note that, $\Delta_i = O_i - O_{i-1} = (O_i - R_{i-1}) + (R_{i-1} - O_{i-1})$. Thus, by the linearity of expectation,

$$\begin{aligned} \mathbb{E} \left[\sum_{i \in [U]} \Delta_i \mid L \right] = \mathbb{E} \left[\sum_{i \in [U]} (O_i - R_{i-1}) \mid L \right] \\ + \mathbb{E} \left[\sum_{i \in [U]} (R_{i-1} - O_{i-1}) \mid L \right]. \end{aligned}$$

We aim to apply Theorem 5 to both expected values on the right-hand side with U and a suitable filtration. To this end, let

$$(M_i)_{i \in \mathbb{N}_{>0}} = (O_{i+1} - R_i)_{i \in \mathbb{N}} \text{ and } (N_i)_{i \in \mathbb{N}_{>0}} = (R_i - O_i)_{i \in \mathbb{N}},$$

and note that the indices of the new sequences start at 1 whereas they start at 0 for the old ones. Note that U is integrable by assumption.

Considering M: Let $(\mathcal{G}_i)_{i \in \mathbb{N}} = (\mathcal{F}_{\inf\{R_i, S\}})_{i \in \mathbb{N}}$. Note that $(M_i \cdot \mathbf{1}(i \leq U))_{i \in \mathbb{N}_{>0}}$ is adapted to \mathcal{G} , as for all $i \in \mathbb{N}_{>0}$, it holds that $R_{i-1} \cdot \mathbf{1}(i \leq U) < O_i \cdot \mathbf{1}(i \leq U) \leq \inf\{R_i, S\}$ and thus that $M_i \cdot \mathbf{1}(i \leq U)$ is \mathcal{G}_i -measurable.

Since $\sum_{i \in [U]} (O_i - O_{i-1}) = \sum_{i \in [U]} M_i + \sum_{i \in [U]} N_i$ is integrable by assumption, and since M and N are non-negative by the definition of O and R , it follows that $\sum_{i \in [U]} M_i$ is integrable. Further, since (1) R_{i-1} is $\mathcal{F}_{R_{i-1}}$ -measurable, and (2) by the assumption that there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}_{>0}$, we have $\mathbb{E}[|O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \leq c$,

$$\begin{aligned} \mathbb{E}[|M_i - \mathbb{E}[M_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \\ = \mathbb{E}[|O_i - R_{i-1} - \mathbb{E}[O_i - R_{i-1} \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \\ \stackrel{(1)}{=} \mathbb{E}[|O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \stackrel{(2)}{\leq} c. \end{aligned}$$

Since, for all $i \in \mathbb{N}_{>0}$ such that $i \leq U$, it holds that $R_{i-1} < S$, it follows for all $i \in \mathbb{N}_{>0}$ that

$$\mathbb{E}[|M_i - \mathbb{E}[M_i \mid \mathcal{G}_{i-1}]| \mid \mathcal{G}_{i-1}] \cdot \mathbf{1}(i - 1 < U) \leq c.$$

Applying Theorem 5 and taking the expectations on both sides yields $\mathbb{E}[\sum_{i \in [U]} M_i \mid L] = \mathbb{E}[\sum_{i \in [U]} \mathbb{E}[M_i \mid \mathcal{G}_{i-1}] \mid L]$.

Considering N: This case is very similar to the previous one. Let $(\mathcal{G}_i)_{i \in \mathbb{N}} = (\mathcal{F}_{\inf\{O_i, S\}})_{i \in \mathbb{N}}$. Note that $(N_i \cdot \mathbf{1}(i \leq U))_{i \in \mathbb{N}_{>0}}$ is adapted to \mathcal{G} , as for all $i \in \mathbb{N}_{>0}$, it holds that $O_{i-1} \cdot \mathbf{1}(i \leq U) \leq R_{i-1} \cdot \mathbf{1}(i \leq U) < \inf\{O_i, S\}$. Further, as in the previous case, since $\sum_{i \in [U]} (O_i - O_{i-1})$ is integrable

by assumption, so is $\sum_{i \in [U]} N_i$. Last, by the assumption that there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}_{>0}$, it holds that $\mathbb{E}[|R_{i-1} - \mathbb{E}[R_{i-1} \mid \mathcal{F}_{O_{i-1}}]| \mid \mathcal{F}_{O_{i-1}}] \leq c$, and since for all $i \in \mathbb{N}_{>0}$ such that $i \leq U$ it holds that $O_{i-1} < S$, it follows for all $i \in \mathbb{N}_{>0}$, by adding $O_{i-1} - O_{i-1}$ inside the expectation, that $\mathbb{E}[|N_i - \mathbb{E}[N_i \mid \mathcal{G}_{i-1}]| \mid \mathcal{G}_{i-1}] \cdot \mathbf{1}(i - 1 < U) \leq c$. Applying Theorem 5 and taking the expectations on both sides yields $\mathbb{E}[\sum_{i \in [U]} N_i \mid L] = \mathbb{E}[\sum_{i \in [U]} \mathbb{E}[N_i \mid \mathcal{G}_{i-1}] \mid L]$.

Concluding: Taking everything together and noting that for all $i \in \mathbb{N}_{>0}$ such that $i \leq U$ it holds that $R_{i-1} < S$ and that $O_{i-1} < S$ concludes the proof. \square

With respect to our setting, utilizing the definitions from equation (12), we get the following result.

Corollary 9 (Translating results from HASMAJORITY to MAJORITY). *Let $n \in \mathbb{N}_{>0}$ be even, let $r \in [n/2]$, and let D be a distribution over $\{0, 1\}^n$. Consider an EA A , represented by the sequence $(\mathbf{x}^{(t)})_{t \in \mathbb{N}}$ with initialization distribution D , adapted to a filtration $(\mathcal{F}_i)_{i \in \mathbb{N}}$.*

Assume that when A optimizes HASMAJORITY $_r$, there is a probability $p_0 \in (0, 1]$ that this solution is also an optimum of MAJORITY. Furthermore, assume the definitions of equation (12).

Let S denote the run time of A on MAJORITY $_r$, and let $U = \sup\{i \in \mathbb{N} \mid O_i \leq S\}$. Assume that $\sum_{i \in [U]} (O_i - O_{i-1})$ and U are integrable as well as that there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}_{>0}$, it holds that $\mathbb{E}[|O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \leq c$ and $\mathbb{E}[|R_{i-1} - \mathbb{E}[R_{i-1} \mid \mathcal{F}_{O_{i-1}}]| \mid \mathcal{F}_{O_{i-1}}] \leq c$.

Last, let L denote the event that $U \geq 1$, and let T denote the run time of A on HASMAJORITY $_r$. Then

$$\begin{aligned} \mathbb{E}[S] = \mathbb{E}[T] + (1 - p_0) \mathbb{E} \left[\sum_{i \in [U]} \left(\mathbb{E}[O_i - R_{i-1} \mid \mathcal{F}_{R_{i-1}}] \right. \right. \\ \left. \left. + \mathbb{E}[R_{i-1} - O_{i-1} \mid \mathcal{F}_{O_{i-1}}] \right) \mid L \right], \end{aligned}$$

where we interpret the second summand to be 0 if $1 - p_0 = 0$.

Corollary 9 states an equality for the expected run time of an EA on MAJORITY. Although calculating the exact expectations of the right-hand side generally seems implausible, the theorem can still be used in order to derive upper and lower bounds for expected run times on MAJORITY by deriving corresponding bounds for all of the expected values in question. Using the notation of the theorem, an easy way for getting a bound on the expectation conditional on L is to derive worst-case bounds for both of the conditional expectations in the sum. If these worst-case bounds are constants, then they can be pulled out of the expectation, and it remains to bound $\mathbb{E}[U \mid L]$, which requires to only reason about the number of times of attempting to find an optimum of MAJORITY when optimizing HASMAJORITY. It is worth noting that the second of the conditional expectations in the sum considers the run time of A optimizing HASMAJORITY when starting with a solution with a majority of 1s. Since this is typically very similar to how T is defined, as only the initialization (and the history of A) differs between both scenarios, this expectation may already follow from a bound on $\mathbb{E}[T]$. Overall, the complexity of the run time analysis for MAJORITY is drastically reduced.

Algorithm 2: The $(1 + 1)$ -unary unbiased algorithm framework with unbiased and memory-restricted mutation operators $(\phi_t)_{t \in \mathbb{N}}$ over $\{0, 1\}^n$ and initial distribution D , maximizing function $f: \{0, 1\}^n \rightarrow \mathbb{R}$. Note for all $t \in \mathbb{N}$ that ϕ_t only has access to a single individual, resulting in a heavy memory restriction.

```

1  $\mathbf{x}^{(0)} \sim D$ ;
2 for  $t \in \mathbb{N}$  do
3    $\mathbf{y} \leftarrow \phi_t(\mathbf{x}^{(t)});$ 
4   if  $f(\mathbf{y}) \geq f(\mathbf{x}^{(t)})$  then  $\mathbf{x}^{(t+1)} \leftarrow \mathbf{y}$ ;
5   else  $\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)};$ 

```

In the following, we apply the technique above in order to derive run time bounds for RLS on MAJORITY. Note that we already have an upper bound on the expected run time on HASMAJORITY for arbitrary initializations, due to [Theorem 6](#). Thus, we are only concerned with how often the algorithms find an optimum \mathbf{x} of HASMAJORITY with a majority of 0s before optimizing MAJORITY, and with how long it takes to get from \mathbf{x} to a solution with a majority of 1s. While the latter question is specific to each algorithm, we answer the former for all unbiased algorithms.

1) *Number of Retries:* Using the notation of [Corollary 9](#), we bound $\mathbb{E}[U \mid D]$ from above for $(1 + 1)$ -unary unbiased elitist algorithms with memory-restriction, as detailed in [Algorithm 2](#). Further, we require the sequence of mutation operator $(\phi_t)_{t \in \mathbb{N}}$ of such an algorithm to be ONEMAX-compliant, which means that each mutation operator satisfies that results of individuals with many 1s are at least as likely to have as many 1s as results of individuals with fewer 1s. Formally, we say that ϕ is ONEMAX-compliant if and only if, for all $t_1, t_2 \in \mathbb{N}$, all $\mathbf{y}, \mathbf{z} \in \{0, 1\}^n$ with $\mathbf{y} = \mathbf{x}^{(t_1)}$, $\mathbf{z} = \mathbf{x}^{(t_2)}$, and $|\mathbf{y}|_1 \leq |\mathbf{z}|_1$, and all $i \in [0..n]$, it holds that $\Pr[|\phi_{t_1}(\mathbf{y})|_1 \geq i \mid \mathbf{y}] \leq \Pr[|\phi_{t_2}(\mathbf{z})|_1 \geq i \mid \mathbf{z}]$. Note that while typical mutation operators are ONEMAX-compliant, operations such as the inversion of a bit string are not. This definition leads to the following result.

Lemma 10. *Assume the setting of [Corollary 9](#) and that $\mathcal{R} := \{O_{i+1} - R_i \mid i \in \mathbb{N} \wedge O_{i+1} \leq S\}$ is independent. If A is an instance of [Algorithm 2](#) with ONEMAX-compliant mutation ϕ , then, conditional on L , it holds that U is stochastically dominated by a geometrically distributed random variable with expectation 2.*

Proof. Note that $|\mathcal{R}| = U$, and recall that $U \geq 1$, as we condition on L .

Let $i \in \mathbb{N}$ such that $M_i := O_{i+1} - R_i$ and $O_{i+1} \leq S$, that is, $M_i \in \mathcal{R}$. We call such an element a *retry*. We say that M_i is a *success* if $O_{i+1} = S$, and a *failure* otherwise. Note that U is the number of retries until the first success and that, by assumption, the retries are mutually independent.

It remains to show that each retry has a probability of at least $1/2$ of being a success. To this end, let $i \in \mathbb{N}$ such that M_i is a retry. First, assume that $|\mathbf{x}^{(R_i)}|_1 = n/2$. Recall that the retry ends once A finds an optimum of HASMAJORITY $_r$. Since A is unbiased, each trajectory of \mathbf{x} that ends with a

solution with a majority of 0s has a symmetric trajectory (in the sense of swapping 0s and 1s) with equal probability that ends with a solution with a majority of 1s. Thus, the probability of M_i being a success is $1/2$.

Now, assume that $|\mathbf{x}^{(R_i)}|_1 > n/2$. Let $\mathbf{y}^{(0)} \in \{0, 1\}^n$ such that $|\mathbf{y}^{(0)}|_1 = n/2$, and, for all $t \in \mathbb{N}$, let $\mathbf{y}^{(t+1)} = \phi_t(\mathbf{y}^{(t)})$. Since ϕ is ONEMAX-compliant, it holds for all $t \in \mathbb{N}$ that $|\mathbf{x}^{(R_i+t)}|_1$ stochastically dominates $|\mathbf{y}^{(t)}|_1$. Thus, there exists a coupling such that for all $t \in \mathbb{N}$, it holds that $|\mathbf{x}^{(R_i+t)}|_1 \geq |\mathbf{y}^{(t)}|_1$ [[20, Theorem 1.8.10](#)]. Thus, the probability of M_i being a success is at least as large as the probability of the process \mathbf{y} finding a solution with at least $n/2 + r$ 1s before finding a solution with at least that many 0s. By the first case above, the probability of this event is $1/2$. This concludes the proof. \square

We note that RLS uses ONEMAX-compliant mutation.

2) *Integrability of the Stopping Time:* An important property to check in [Corollary 9](#) is that $\sum_{i \in [U]} (O_i - O_{i-1})$ is integrable. To this end, it suffices to show that this random variable has *some* finite expectation, as the sum is non-negative. We do so by applying [Theorem 1](#).

Lemma 11. *Assume the setting of [Corollary 9](#). Let $(\mathcal{F}_t)_{t \in \mathbb{N}}$ denote the natural filtration of \mathbf{x} , and assume that there is a $p \in (0, 1]$ such that, for all $t \in \mathbb{N}$, it holds that*

$$\Pr[|\mathbf{x}^{(t+1)}|_1 > |\mathbf{x}^{(t)}|_1 \mid \mathcal{F}_t] \cdot \mathbb{1}(t < S) \geq p \cdot \mathbb{1}(t < S). \quad (13)$$

Then $\sum_{i \in [U]} (O_i - O_{i-1})$ is integrable.

Proof. We aim to prove that $\mathbb{E}[S]$ is integrable via [Theorem 1](#). Since $\sum_{i \in [U]} (O_i - O_{i-1}) \leq S$ and the terms of the sum are non-negative, the claim then follows. Recall that we assume that $n \geq 2$.

Let $(X_t)_{t \in \mathbb{N}} = ((n/p)^n - (n/p)^{|\mathbf{x}^{(t)}|_1})$, and note that X is non-negative and adapted to \mathcal{F} . Further note that [inequality \(1\)](#) is satisfied for all $t \in \mathbb{N}$, since, using [equation \(13\)](#), $n \geq 2$, and $p \leq 1$, it follows that

$$\begin{aligned} & (X_t - \mathbb{E}[X_{t+1} \mid \mathcal{F}_t]) \cdot \mathbb{1}(t < T) \\ &= \left(\binom{n}{p}^{|\mathbf{x}^{(t+1)}|_1} - \binom{n}{p}^{|\mathbf{x}^{(t)}|_1} \right) \\ & \quad \cdot \Pr[|\mathbf{x}^{(t+1)}|_1 < |\mathbf{x}^{(t)}|_1 \mid \mathcal{F}_t] \cdot \mathbb{1}(t < S) \\ & \quad + \left(\binom{n}{p}^{|\mathbf{x}^{(t+1)}|_1} - \binom{n}{p}^{|\mathbf{x}^{(t)}|_1} \right) \\ & \quad \cdot \Pr[|\mathbf{x}^{(t+1)}|_1 > |\mathbf{x}^{(t)}|_1 \mid \mathcal{F}_t] \cdot \mathbb{1}(t < S) \\ & \geq \left(1 - \binom{n}{p}^{|\mathbf{x}^{(t)}|_1} \right) \cdot \mathbb{1}(t < S) \\ & \quad + \binom{n}{p}^{|\mathbf{x}^{(t)}|_1} \cdot \left(\frac{n}{p} - 1 \right) \cdot p \cdot \mathbb{1}(t < S) \\ & = \left(1 + \binom{n}{p}^{|\mathbf{x}^{(t)}|_1} \cdot \left(\frac{n}{p} - 2 \right) \right) \cdot \mathbb{1}(t < S) \\ & \geq \mathbb{1}(t < S). \end{aligned}$$

Applying [Theorem 1](#) with $\delta = 1$, noting that $\mathbb{E}[X_S] \geq 0$, and then taking the expectation yields a finite bound for $\mathbb{E}[S]$,

concluding the proof. \square

3) *Reaching a Majority of Ones:* The only thing left in order to apply [Corollary 9](#) is to bound the expected time it takes an algorithm to get from a solution with a majority of 0s to one with a majority of 1s.

Theorem 12. *Consider RLS optimizing MAJORITY₀, and assume that the initial individual $\mathbf{x}^{(0)}$ has at least $n/2 + 1$ 0s. Then the expected run time of RLS is at most $n(1 + \ln(|\mathbf{x}^{(0)}|_0 - n/2))/2$.*

Proof. We aim to apply [Theorem 2](#). To this end, let $(\mathbf{x}^{(t)})_{t \in \mathbb{N}}$ denote the best individual of RLS after each iteration, and let $(X_t)_{t \in \mathbb{N}}$ be such that for all $t \in \mathbb{N}$, it holds that $X_t = |\mathbf{x}^{(t)}|_0 - n/2$. Further, let \mathcal{F} be the natural filtration of \mathbf{x} , and let $T = \inf\{t \mid X_t < 1\}$. Note that T denotes the run time of RLS on MAJORITY₀⁽¹⁾.

Since $X_0 \geq 1$ by assumption, we are left to check [inequality \(2\)](#). Let $t \in \mathbb{N}$, and assume that $t < T$. Note that then

$$\begin{aligned} X_t - \mathbb{E}[X_{t+1} \mid \mathcal{F}_t] &= -1 \cdot \frac{n - |\mathbf{x}^{(t)}|_0}{n} + 1 \cdot \frac{|\mathbf{x}^{(t)}|_0}{n} \\ &= \frac{2|\mathbf{x}^{(t)}|_0 - n}{n} = \frac{2}{n}X_t. \end{aligned}$$

Applying [Theorem 2](#) with $\delta = 2/n$ concludes the proof. \square

4) *Deriving the Run Time Result:* By combining the results from the previous sections, the run time bound on MAJORITY follows straightforwardly.

Proof of Theorem 7. We aim to apply [Corollary 9](#). To this end, we check the assumptions of the theorem and use its notation.

Due to the uniform initialization and the symmetry of the mutation of RLS (with respect to the number of 0s and 1s), it holds that $p_0 = 1/2$. By [Lemma 11](#), $\sum_{i \in [U]} (O_i - O_{i-1})$ is integrable, and by [Lemma 10](#), using that $\mathbb{E}[U] \leq \mathbb{E}[U \mid L] + \mathbb{E}[U \mid \bar{L}]$ and, by definition of L , that $\mathbb{E}[U \mid \bar{L}] = 0$, it follows that U is integrable. The last condition to check is that there is a $c \in \mathbb{R}$ such that for all $i \in \mathbb{N}_{>0}$, it holds that $\mathbb{E}[|O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \leq c$ and $\mathbb{E}[|R_{i-1} - \mathbb{E}[R_{i-1} \mid \mathcal{F}_{O_{i-1}}]| \mid \mathcal{F}_{O_{i-1}}] \leq c$. Let $i \in \mathbb{N}_{>0}$. For the first inequality, we note that, by the triangle inequality and Jensen's inequality,

$$\begin{aligned} &\mathbb{E}[|O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \\ &= \mathbb{E}[|O_i - R_{i-1} - \mathbb{E}[O_i - R_{i-1} \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \\ &\leq 2\mathbb{E}[|O_i - R_{i-1}| \mid \mathcal{F}_{R_{i-1}}]. \end{aligned}$$

By [Theorem 6](#), using its notation, it holds that

$$\mathbb{E}[|O_i - R_{i-1}| \mid \mathcal{F}_{R_{i-1}}] \leq 3r \cdot h(\mathbf{x}^{(R_{i-1})})/(\lambda - 1).$$

Since, for all $\mathbf{x} \in \{0, 1\}^n$, it holds that $h(\mathbf{x}) \leq \lambda^r - 1$, it follows overall that $\mathbb{E}[|O_i - \mathbb{E}[O_i \mid \mathcal{F}_{R_{i-1}}]| \mid \mathcal{F}_{R_{i-1}}] \leq 6r \cdot (\lambda^r - 1)/(\lambda - 1) =: c_1$. Similarly, we see that

$$\begin{aligned} &\mathbb{E}[|R_{i-1} - \mathbb{E}[R_{i-1} \mid \mathcal{F}_{O_{i-1}}]| \mid \mathcal{F}_{O_{i-1}}] \\ &= \mathbb{E}[|R_{i-1} - O_{i-1} - \mathbb{E}[R_{i-1} - O_{i-1} \mid \mathcal{F}_{O_{i-1}}]| \mid \mathcal{F}_{O_{i-1}}] \\ &\leq 2\mathbb{E}[|R_{i-1} - O_{i-1}| \mid \mathcal{F}_{O_{i-1}}]. \end{aligned}$$

By [Theorem 12](#), it holds that

$$\mathbb{E}[|R_{i-1} - O_{i-1}| \mid \mathcal{F}_{O_{i-1}}] \leq \frac{n}{2}(1 + \ln(|\mathbf{x}^{(O_{i-1})}|_0 - n/2)).$$

Noting that $|\mathbf{x}^{(O_{i-1})}|_0 = n/2 + r$, since RLS cannot get to a solution with strictly more than $n/2 + r$ 0s before getting exactly $n/2 + r$ 0s, we get $\mathbb{E}[|R_{i-1} - \mathbb{E}[R_{i-1} \mid \mathcal{F}_{O_{i-1}}]| \mid \mathcal{F}_{O_{i-1}}] \leq n(1 + \ln(r)) =: c_2$. Thus, choosing $c = \max\{c_1, c_2\}$ satisfies the remaining condition of [Corollary 9](#).

We continue with bounding $\mathbb{E}[T]$, noting that for all $i \in \mathbb{N}_{>0}$ such that $i \in [U]$, we bounded the expectations $\mathbb{E}[O_i - R_{i-1} \mid \mathcal{F}_{R_{i-1}}]$ and $\mathbb{E}[R_{i-1} - O_{i-1} \mid \mathcal{F}_{O_{i-1}}]$ already above, as the differences are non-negative and the absolute value thus does not change anything.

Let $i \in \mathbb{N}_{>0}$ such that $i \in [U]$. Note that $O_i - R_{i-1}$ and T have the same distribution conditional on the same initial individual for the respective time interval, since RLS is Markovian and since both stopping times stop once an optimum of HASMAJORITY_r is found. Thus, their conditional expectations (on the same initial individual) are the same, and [Theorem 6](#) yields a bound for $\mathbb{E}[T]$, too.

Combining all bounds, by the tower rule for expectation, [Corollary 9](#) yields

$$\begin{aligned} \mathbb{E}[S] &\leq 3r \cdot \frac{\lambda^r - 1}{\lambda - 1} \\ &\quad + \frac{1}{2} \mathbb{E} \left[\sum_{i \in [U]} \left(n \frac{1 + \ln(r)}{2} + 3r \cdot \frac{\lambda^r - 1}{\lambda - 1} \right) \mid L \right] \\ &= 3r \cdot \frac{\lambda^r - 1}{\lambda - 1} + \frac{1}{2} \left(n \frac{1 + \ln(r)}{2} + 3r \cdot \frac{\lambda^r - 1}{\lambda - 1} \right) \mathbb{E}[U \mid L]. \end{aligned}$$

Applying [Lemma 10](#) yields $\mathbb{E}[U \mid L] \leq 2$, thus concluding the proof. \square

C. Results for Neutrality in the W-Model

For all $n \in \mathbb{N}_{>0}$, all $\mathbf{x} \in \{0, 1\}^n$, and all $S \subseteq [n]$, let \mathbf{x}_S denote the bit string of length $|S|$ that consists only of those entries in \mathbf{x} with an index in S , that is, $\mathbf{x}_S = (\mathbf{x}_i)_{i \in S}$.

Let $n \in \mathbb{N}_{>0}$. Given a pseudo-Boolean function $f: \{0, 1\}^n \rightarrow \mathbb{R}$ as well as $k \in \mathbb{N}_{>0}$, the property of neutrality in the W-model [1] constructs a new function $f': \{0, 1\}^{nk} \rightarrow \mathbb{R}$ with

$$\mathbf{x} \mapsto f(\text{MAJORITY}_1(\mathbf{x}_{[(i-1)k+1..ik]})_{i \in [n]}).$$

In other words, each bit of f is exchanged for a block of k bits. When evaluating f' , for each block, the majority of bits is determined, and then f is evaluated on the string of the n majority bits. We refer to f' as d with added neutrality of degree k .

Applying the results by Doerr et al. [2], we get the following bound for the expected run time of RLS on the pseudo-Boolean function ONEMAX: $\mathbf{x} \mapsto |\mathbf{x}|_1$ with added neutrality.

Corollary 13. *Let $n, k \in \mathbb{N}_{>0}$ such that n is even. For sufficiently large n , the expected run time of RLS on ONEMAX with added neutrality of degree k is at most $O(nk \log(n))$.*

Before we state the result by Doerr et al. [2] and prove [Corollary 13](#), we introduce some notation of their paper to the degree that it is necessary for our result. Let $n, k \in \mathbb{N}_{>0}$ and

$i \in [n]$, we say that a function $g: \{0, 1\}^{nk} \rightarrow \mathbb{R}$ acts only on block i if and only if for all $\mathbf{x} \in \{0, 1\}^{nk}$, it holds that $g(\mathbf{x}) = g(\mathbf{x}_{\{(i-1)k+1..ik\}})$. We say that a function $f: \{0, 1\}^{nk}$ is a consecutively separable function composed of n sub-functions if and only if there exists a collection $(g_i)_{i \in [n]}$ of n pseudo-Boolean functions, each of dimension nk , such that for all $\mathbf{x} \in \{0, 1\}^{nk}$, it holds that $f(\mathbf{x}) = \sum_{i \in [n]} g_i(\mathbf{x})$, and for all $i \in [n]$, it holds that g_i acts only on block i .

Theorem 14 (Simplification of [2, Theorem 2]). *Let $n, k \in \mathbb{N}_{>0}$, and let $f: \{0, 1\}^{nk} \rightarrow \mathbb{R}$ be a consecutively separable function composed of n sub-functions. Further, let T^* denote the maximum expected run time of RLS with uniform initialization over all sub-functions. Then the expected run time of RLS with uniform initialization on f is $O(T^* \log(n))$.*

Proof of Corollary 13. Note that ONEMAX with added neutrality of degree k is a consecutively separable function composed of n sub-functions, each of which acts only on a single block of length k as MAJORITY₁. Thus, we aim to apply Theorem 14 and determine T^* by Theorem 7, noting that each block has the same sub-function. However, note that Theorem 7 is not directly applicable, since it considers MAJORITY₁ to be defined over the entire search space (that is, for all bits), whereas in the setting of Corollary 13, MAJORITY₁ is a sub-function that only acts on a block of length k . Hence, we translate the result from Theorem 7 to our setting. We do so by only counting those mutations that change a bit in block i and adding waiting time due to iterations that change other (irrelevant) bits for a specific sub-function.

Formally, let $i \in [n]$, and consider the MAJORITY₁ sub-function g_i acting only on block i . Further, let S denote the expected run time of RLS with uniform initialization on MAJORITY₁ defined over bit strings of length k , and let T denote the expected run time of RLS with uniform initialization on g_i . We note that, similar to the proof of Lemma 11, T is integrable, allowing us to reorder terms when calculating $E[T]$. Last, let $(\mathbf{x}^{(t)})_{t \in \mathbb{N}}$ denote the trajectory of RLS with uniform initialization on g_i , and for all $t \in \mathbb{N}_{>0}$, let M_t denote the event that $\mathbf{x}^{(t)}$ is a result of a mutation that changes a bit in block i . We couple S and T such that

$$\sum_{t \in [T]} \mathbb{1}(M_t) = S, \quad (14)$$

that is, only considering changes in block i , the optimization behaves as if considering the optimization of MAJORITY₁ defined of bit strings of length k . Further, let $I = \{t \in [T] \mid \mathbb{1}(M_t) = 1\}$.

Since it trivially holds that $T = \sum_{t \in [T]} (\mathbb{1}(M_t) + \mathbb{1}(\overline{M_t}))$, by the definition of I , it follows that $T = \sum_{t \in I} \mathbb{1}(M_t) + \sum_{[T] \setminus I} \mathbb{1}(\overline{M_t})$. We aim to connect the second sum to the first, since we know how to bound the first sum by Equation (14). To this end, for all $t \in I$, let $J_t = \{t' \in [T] \mid t' \leq t \wedge \forall s \in I, s < t: s < t'\}$, that is, the set of all time points that are in between the time points in I . Note that $\mathbb{1}(M_T) = 1$ and, thus, $T \in I$, since the final mutation has to change a bit in block i . Thus, $[T] = \bigcup_{t \in I} J_t$. Further note for all $t \in I$ that $|J_t|$ follows a geometric distribution with support $\mathbb{N}_{>0}$ and with success probability $k/(nk) = 1/n$, since the mutation of RLS

chooses uniformly at random which bit to flip. In addition, the $\{|J_t|_{t \in I}\}$ are independently and identically distributed. Let J denote their common distribution.

Using the definitions of $\{|J_t|_{t \in I}\}$ and J as well as equation (14), we see that $T = \sum_{t \in I} |J_t| \cdot \mathbb{1}(M_t) = J \cdot S$. Since J and S are independent, as the mutation of RLS is independent of any random choice besides which bit it flips, it follows that $E[J \cdot S] = E[J] \cdot E[S]$. By definition, it follows that $E[J] = n$, and by Theorem 7, it follows that $E[S] \leq 6 + k/2$. Thus, $E[T] = O(nk)$.

Overall, it follows that $T^* = O(nk)$. Applying Theorem 14 concludes the proof. \square

IV. EMPIRICAL RESULTS FOR RLS_ℓ

We complement our theoretical results on MAJORITY with empirical investigations on a generalization of RLS called RLS_ℓ, which flips exactly $ℓ$ bits each iteration. In more detail, RLS_ℓ follows Algorithm 2 and uses a time-homogeneous mutation operator, that is, it uses the same mutation operator in each iteration. Let $n \in \mathbb{N}_{>0}$, and let $\mathbf{x} \in \{0, 1\}^n$. Given an $ℓ \in [n]$, for all $t \in \mathbb{N}$, RLS_ℓ computes the result of $\phi_t(\mathbf{x})$ by first choosing a subset $I \subseteq [n]$ of cardinality $ℓ$ uniformly at random among all $ℓ$ -size subsets of $[n]$ and then flipping exactly the bits at the positions in I . Note that RLS is a special case of RLS_ℓ for $ℓ = 1$.

In our experiments, we analyze the impact of the parameter $ℓ$ of RLS_ℓ on the run time for MAJORITY when using the uniform distribution as initialization distribution. To this end, given a problem size of $n \in \mathbb{N}_{>0}$, with even n , we fix the parameter r of MAJORITY_r to $\lfloor \sqrt{n} \rfloor$. Recalling our discussion of r after Theorem 7, this results in an expected run time linear in n for HASMAJORITY_r, which is a lower bound for the run time for MAJORITY (see Corollary 9). For larger values of r , the expected run time increases drastically. Thus, $r = \lfloor \sqrt{n} \rfloor$ results in an easily-solvable problem that is not yet trivial. We note though that, due to the uniform initialization, there is a constant probability of the initial solution being already optimal.

A. Near-optimal Value of $ℓ$

We determine the value $ℓ$ of RLS_ℓ that has the best average empirical run time among a variety of different values of $ℓ$. For each problem size $n \in \{10^i \mid i \in [2..5]\} \cup \{5 \cdot 10^i \mid i \in [2..4]\}$, we consider $ℓ \in \{1, 2, 3, 4, \lfloor \ln n \rfloor, \lfloor \sqrt{n} \rfloor\} \cup \{\lfloor an/12 \rfloor \mid a \in \{3\} \cup [6..10]\}$. For each parameter combination of n and $ℓ$, we start 1000 independent runs of RLS_ℓ (on MAJORITY _{$\lfloor \sqrt{n} \rfloor$}) and store its empirical run time, that is, the number of iterations t from Algorithm 2 until an optimal solution was found for the first time. Afterward, we compute the average, median, and the 25th as well as 75th percentile. Figure 5 shows the results for $n \in \{100, 50\,000, 100\,000\}$.

The results for our values of n look qualitatively similarly, that is, the run time drastically decreases by several orders of magnitude with increasing $ℓ$ until $ℓ \in \Theta(n)$. The median is always below the mean, and the area between the 25th percentile and the mean is larger than the one between the mean and the 75th percentile. This is likely due to the initialization creating an optimal solution or one that is close to an optimal

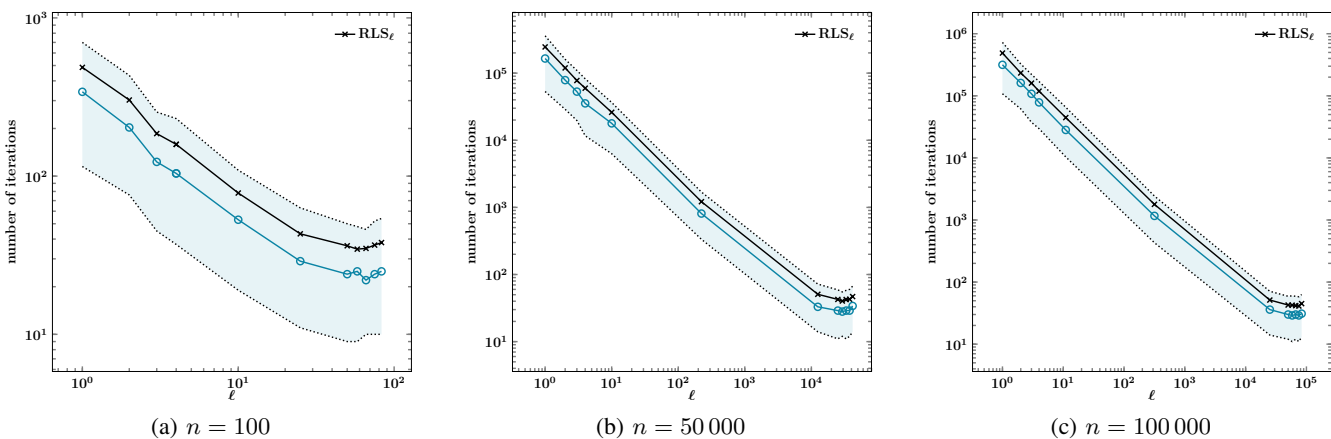


Fig. 5: The empirical run time (number of iterations until an optimum was found) of RLS_ℓ on $MAJORITY_{\lfloor \sqrt{n} \rfloor}$ for $n \in \{100, 50\,000, 100\,000\}$, each with varying values of ℓ , as discussed at the beginning of Section IV-A. Note the doubly logarithmic scale. For each combination of n and ℓ , 1000 independent runs with uniform initialization were started. The solid black line (points represented by \times) depicts the average of the runs. The solid blue line (points represented by \circ) depicts the median of the runs, and the surrounding shaded area depicts the range from the 25th percentile to the 75th percentile.

one with respect to its number of 1s. Such runs result in a low run time. The mean being larger than the median suggests that initial solutions that are further away from optimal solutions in terms of their number of 1s result in a far larger run time.

In the regime of $\ell \in \{\lfloor an/12 \rfloor \mid a \in \{3\} \cup [6..10]\}$, the minimum average empirical run time is always taken for one of the values $\ell \in \{\lfloor an/12 \rfloor \mid a \in [6..9]\}$. For these values of ℓ , the difference of the smallest average to the second smallest is typically less than 1. This suggests that, in this regime, the run time of RLS_ℓ does not depend heavily on ℓ . However, overall, a linear dependence of ℓ on n seems best.

B. Trajectories for Different Values of ℓ

The results in Figure 5 show that the value of ℓ has a drastic impact on the run time of RLS_ℓ on $MAJORITY_{\lfloor \sqrt{n} \rfloor}$. Since ℓ only influences how largely the mutation changes the current solution but does not influence the selection of which solution to use for the next iteration, we compare the trajectories of RLS_ℓ for different values of ℓ . We choose $n = 10\,000$ and $\ell \in \{1, \lfloor \ln n \rfloor, \lfloor n/2 \rfloor\}$, and we log at each iteration the number of 1s of the currently best solution (the *fitness level*). Note that all optimal solutions are at fitness levels at least 5100.

Figure 6 depicts our results. Note that there is a huge difference in run time between $\ell = 1$ and $\ell = \lfloor \ln n \rfloor$, which is most likely due to the former starting with a solution of more than $5000 = n/2$ 1s, whereas the latter starts at a position almost at $n/2 - \sqrt{n}$. This highlights the impact of the initialization on the run time. Interestingly, the case $\ell = \lfloor n/2 \rfloor$ also starts with a rather bad solution of about $n/2 - \sqrt{n}/2$ 1s but manages to quickly find an optimum. In fact, the last mutation increases the number of 1s by about 200. This suggests that RLS_ℓ has far better capabilities of exploring the plateau with $\ell \in \Theta(n)$ than with smaller values of ℓ .

V. CONCLUSION

We analyzed the expected run time that RLS, started on a plateau, requires until it leaves the plateau for the first time. We

considered a symmetric (HASMAJORITY) and an asymmetric (MAJORITY) setting, and we showed, for general EAs, how to derive a result for MAJORITY from a result on HASMAJORITY (Corollary 9), based on a more general result for a random process to reach a certain offset (Theorem 8). A fair amount of conditions of Corollary 9 follow directly from Lemmas 10 and 11, which apply to many EAs. Lemma 11 is so general that it holds for any EA that has, for each individual, a positive probability generating it. And although Lemma 10 is not as easy to check, we note that it holds for the $(1+1)$ EA [21, Lemma 6.1].

Theorem 8 provides a tool for analyzing *general* plateaus by considering phases in which the algorithm tries to cross the plateau. In order to bound the expected number of restarts, one determines the probability of crossing the plateau and not returning to its start. This can be done by defining a drift potential that decreases toward the end of the plateau (as done in the proof of Theorem 6) but is also 0 at the start. The optional-stopping theorem allows then to determine the desired probability. However, we note that simply using the potential from the proof of Theorem 6 is typically a too coarse estimate since the bound always assumes the worst case of crossing the plateau. It remains an interesting open problem to derive tighter bounds for such a setting.

Further possible future work includes deriving a lower bound for RLS on HASMAJORITY, as well as analyzing the expected run time of more algorithms on HASMAJORITY, such as the $(1+1)$ EA, RLS_ℓ , or non-elitist EAs. Any such bound translates almost directly into a bound for MAJORITY. Similarly, combining our results with those of Doerr et al. [2] on separable functions provides good bounds for RLS and the $(1+1)$ EA on ONEMAX functions with any degree of neutrality from the W-model. Extending the results of Doerr et al. [2] to other algorithms or to functions other than ONEMAX also helps extend the picture about how well EAs cope with neutrality. Last, the current definitions of HASMAJORITY and MAJORITY let the algorithm either start on a plateau or in a global optimum.

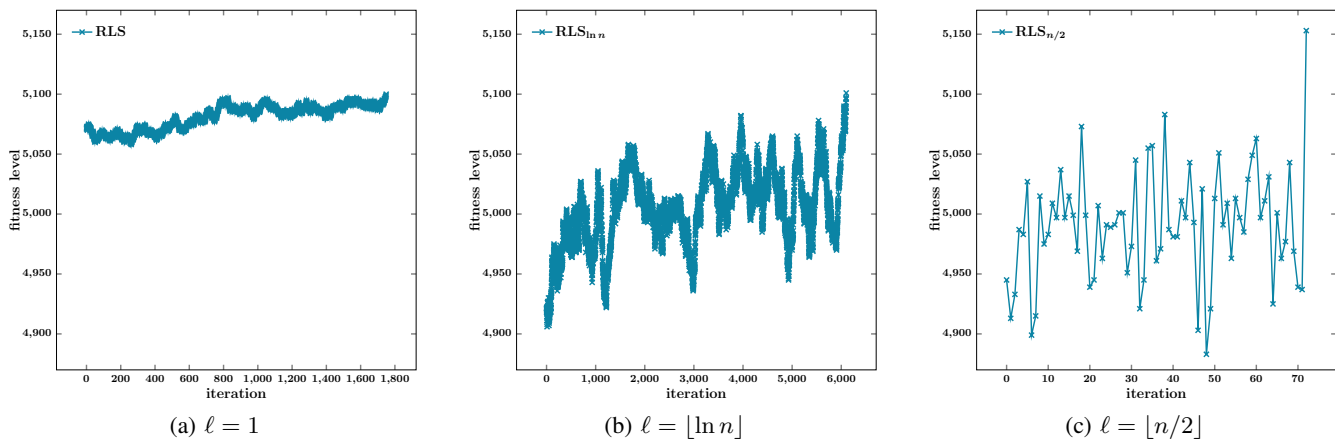


Fig. 6: The number of 1s (*fitness level*) of the best solution of RLS_ℓ on $MAJORITY_{\lfloor\sqrt{n}\rfloor}$ for $n = 10000$ and for different values of ℓ . Each plot depicts a single run that was stopped once an optimal solution was found (fitness level of at least 5100). Note that we show runs that started from a non-optimal solution, chosen uniformly at random among all possible 2^n solutions.

A potential generalization is to place the plateau somewhere else in the search space and introduce an easy slope toward the plateau. This follows the same idea as the function PLATEAU by Antipov and Doerr [5] but generalizing it even further such that the function has more than a single optimum.

Our long-term objective is to analyze the impact of the other components of the W-model problem generator proposed by Weise et al. [1] and extended by Doerr et al. [22]—first individually for each module (as done here for *neutrality*) and then for combinations of the four layers (*neutrality*, *dummy variables*, *epistasis*, and *ruggedness*). We see this as an important step towards run time results that more explicitly link algorithms’ performance to problem characteristics.

REFERENCES

- [1] T. Weise, Y. Chen, X. Li, and Z. Wu, “Selecting a diverse set of benchmark instances from a tunable model problem for black-box discrete optimization algorithms,” *Applied Soft Computing*, vol. 92, p. 106269, 2020.
- [2] B. Doerr, D. Sudholt, and C. Witt, “When do evolutionary algorithms optimize separable functions in parallel?” in *Proc. of FOGA’13*, 2013, pp. 51–64.
- [3] C. Bian, C. Qian, K. Tang, and Y. Yu, “Running time analysis of the (1+1)-EA for robust linear optimization,” *Theoretical Computer Science*, vol. 843, pp. 57–72, 2020.
- [4] J. Garnier, L. Kallel, and M. Schoenauer, “Rigorous hitting times for binary mutations,” *Evol. Comput.*, vol. 7, no. 2, pp. 173–203, 1999.
- [5] D. Antipov and B. Doerr, “Precise runtime analysis for plateau functions,” *ACM Trans. Evol. Learn. Optim.*, vol. 1, pp. 13:1–13:28, 2021.
- [6] H. Bambury, A. Bultel, and B. Doerr, “Generalized jump functions,” in *Proc. of GECCO’21*, 2021, pp. 1124–1132.
- [7] D. Antipov and S. Naumov, “The effect of non-symmetric fitness: the analysis of crossover-based algorithms on RealJump functions,” in *Proc. of FOGA’21*, 2021, pp. 10:1–10:15.
- [8] C. Witt, “On crossing fitness valleys with majority-vote crossover and estimation-of-distribution algorithms,” in *Proc. of FOGA’21*, 2021, pp. 2:1–2:15.
- [9] D.-C. Dang, T. Friedrich, T. Kötzing, M. S. Krejca, P. K. Lehre, P. S. Oliveto, D. Sudholt, and A. M. Sutton, “Escaping local optima with diversity mechanisms and crossover,” in *Proc. of GECCO’16*, 2016, pp. 645–652.
- [10] —, “Escaping local optima using crossover with emergent diversity,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 3, pp. 484–497, 2018.
- [11] L. D. Whitley, S. Varadarajan, R. Hirsch, and A. Mukhopadhyay, “Exploration and exploitation without mutation: Solving the jump function in $\sqrt{\vartheta}$ time,” in *Proc. of PPSN XV*, 2018, pp. 55–66.
- [12] J. Lengler, “Drift analysis,” in [23]. Springer, 2020, pp. 89–131, also available at <https://arxiv.org/abs/1712.00964>.
- [13] J. He and X. Yao, “Drift analysis and average time complexity of evolutionary algorithms,” *Artificial Intelligence*, vol. 127, pp. 57–85, 2001.
- [14] B. Doerr, D. Johannsen, and C. Winzen, “Multiplicative drift analysis,” *Algorithmica*, vol. 64, no. 4, pp. 673–697, 2012.
- [15] T. Kötzing and M. S. Krejca, “First-hitting times under drift,” *Theoretical Computer Science*, vol. 796, pp. 51–69, 2019.
- [16] R. Durrett, *Probability: theory and examples*. Cambridge University Press, 2019.
- [17] Y. Kovchegov. Mth 664: Lectures 24 - 27. Accessed 2021-10-28. [Online]. Available: http://sites.science.oregonstate.edu/~kovchegy/math664winter2013/664_lectures24-27.pdf
- [18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. MIT Press, 2009.
- [19] T. Friedrich, A. Göbel, N. Klodt, M. S. Krejca, and M. Pappik, “Analysis of the survival time of the SIS and SIRS process on stars and cliques,” *CoRR*, vol. abs/2205.02653, 2022. [Online]. Available: <http://arxiv.org/abs/2205.02653>
- [20] B. Doerr, “Probabilistic tools for the analysis of randomized optimization heuristics,” in [23]. Springer, 2020, pp. 1–87, also available at <https://arxiv.org/abs/1801.06733>.
- [21] C. Witt, “Tight bounds on the optimization time of a randomized search heuristic on linear functions,” *Combinatorics, Probability and Computing*, vol. 22, no. 2, pp. 294–318, 2013.
- [22] C. Doerr, F. Ye, N. Horesh, H. Wang, O. M. Shir, and T. Bäck, “Benchmarking discrete optimization heuristics with iohprofiler,” *Applied Soft Computing*, vol. 88, p. 106027, 2020.
- [23] B. Doerr and F. Neumann, Eds., *Theory of Evolutionary Computation: Recent Developments in Discrete Optimization*, 1st ed. Springer, 2020.

Carola Doerr, formerly Winzen, is since 2013 a permanent researcher at Sorbonne Université in Paris, France. Carola obtained her PhD in Computer Science from Saarland University and the Max Planck Institute for Informatics in 2011, and she successfully defender her habilitation (HDR) at Sorbonne Université in 2020. She works on theoretical analysis, benchmarking, and practical applications of black-box optimization heuristics.

Martin S. Krejca obtained his PhD from the Hasso Plattner Institute, University of Potsdam, Germany, in 2019. Since 2022, he is an assistant professor at Ecole Polytechnique, Palaiseau, France. His research interests are the theoretical analysis of random processes, especially black-box optimization heuristics.